

## The Claims

1. (Currently Amended) One or more computer-readable memories comprising computer-executable instructions that, when executed, direct a processor to perform acts comprising:

automatically identifying a custom field on a source code form definition, which defines a form to be generated, and one or more restrictions on an input to the custom field;

~~automatically identifying validation code that, when executed, validates that the input conforms to the one or more restrictions;~~

automatically identifying, on the source code from definition, a custom tag corresponding to the custom field, wherein the custom tag includes an indication of one or more attributes, and wherein each of the one or more attributes includes a value indicating what input corresponding to the non-custom field is to be restricted to;  
and

identifying, from a plurality of pieces of validation code, the validation code corresponding to the one or more attributes of the custom tag; and

adding, to a new form definition that includes a non-custom field corresponding to the custom field, the identified validation code

replacing the custom tag with a tag for the non-custom field and executable code to generate the validation code for the tag for the non-custom field.

2. (Previously Presented) One or more computer-readable memories as recited in claim 1, wherein the computer-executable instructions further direct the processor to perform acts comprising:

adding, to the new form definition, a reference to the identified validation code that, when executed by another processor, causes the other processor to execute the identified validation code.

3. (Previously Presented) One or more computer-readable memories as recited in claim 1, wherein the automatically identifying validation code comprises identifying pre-defined validation code.

4. (Previously Presented) One or more computer-readable memories as recited in claim 1, wherein the source code form definition that defines the custom field includes a tag corresponding to the custom field.

5. (Previously Presented) One or more computer-readable memories as recited in claim 1, wherein the input comprises a user input.

Claim 6: Canceled

7-18. (Canceled)

19. (Currently Amended) A computerized method, comprising:  
automatically identifying, from an input form definition written in a source code, one or more desired fields to be included on a form to be generated via which data can be input; and

~~automatically adding validation code to source code of the form to be generated, wherein the validation code is based at least in part on the one or more desired fields and one or more desired input restrictions associated with the one or more desired fields~~

automatically identifying a custom field tag corresponding to the one or more fields to be included on the form to be generated from the source code, wherein the custom field tag includes an indication of one or more attributes, each of which including a value indicating one or more desired input restrictions associated with the one or more desired fields; and

replacing the custom tag with a non-custom field tag and executable code to generate validation code to enforce the one or more desired input restriction in the one or more desired fields.

Claim 20: Canceled

21. (Original) A method as recited in claim 19, wherein the input comprises a user input.

22. (Currently Amended) A method as recited in claim 19, wherein the automatically ~~adding~~ identifying a custom field tag comprises:

generating a temporary form definition;

adding execution code to the temporary form definition;

executing the execution code to add the validation code to the temporary form definition; and

outputting, as the source code, the temporary form definition.

23. (Currently Amended) A computing system comprising:

a processor, configured to generate[,,];

a form analyzer configured to automatically identify one or more custom tags in a source code form definition, which defines a form to be created; and

a tag replacement module, coupled to the form analyzer, configured to automatically replace each of the one or more custom tags with another tag, and further to add, to a form definition, for each of the one or more custom tags, executable code to generate validation code to validate subsequent inputs to a field corresponding to the tag in the form to be created.

24. (Original) A system as recited in claim 23, wherein the inputs comprise user inputs.

25. (Original) A system as recited in claim 23, wherein the system comprises a compiler.

26. (Original) A system as recited in claim 23, wherein each of the other tags with which the tag replacement module replaces a custom tag is a HyperText Markup Language (HTML) tag.

27. (Original) A system as recited in claim 23, wherein the tag replacement module is further configured to add a reference to the added validation code.

28. (Previously presented) A system as recited in claim 23, wherein the tag replacement module is further configured to generate a new document corresponding to the form definition, to replace each of the one or more custom tags with another tag by adding the other tag to the new document, and to add validation code by adding the validation code to the new document.

29. (Original) A system as recited in claim 23, wherein a plurality of the one or more custom tags have restrictions corresponding to the same validation code, and wherein the tag replacement module is further configured to add the same validation code only once.

30. (Original) A system as recited in claim 23, further comprising a tag library, coupled to the tag replacement module, to store the validation code.

31. (Original) A system as recited in claim 30, wherein the tag library is further to store an identification of the one or more custom tags.

32. (Previously Presented) A computerized method comprising:  
receiving a form definition, written in source code defining a form to be generated, including one or more custom tags, wherein each custom tag corresponds to a data input, and wherein each custom tag includes one or more associated input restrictions; and  
for each of the one or more custom tags,  
automatically identifying a replacement non-custom tag,  
automatically adding the identified replacement non-custom tag to a new form definition,  
automatically identifying validation code that, when executed based on an input corresponding to the tag, validates whether the associated input restrictions are satisfied, and  
automatically adding the identified validation code to the new form definition, such that a user input in a form created from the new form definition is validated.

33. (Previously Presented) A computerized method as recited in claim 32, wherein the method further comprises, for each of the one or more custom tags:  
adding, to the new form definition, a reference to invoke the added validation code.

34. (Previously Presented) A computerized method as recited in claim 32, wherein the receiving further comprises receiving, as part of the form definition, one or more non-custom tags, and wherein the method further comprises adding each of the non-custom tags to the new form definition.

35. (Previously Presented) A computerized method as recited in claim 32, wherein the data input comprises data input by a user.

36. (Previously Presented) A computerized method as recited in claim 32, wherein each input custom tag includes one or more attributes that identify the one or more associated input restrictions, and wherein each of the one or more attributes includes an indication of the attribute and a corresponding value that data input corresponding to the tag is to be restricted to.

37. (Previously Presented) A computerized method as recited in claim 32, wherein automatically adding the identified validation code comprises:

adding execution code to the new form definition; and

executing the execution code to add the identified validation code to the new form definition.

38. (Previously Presented) One or more computer-readable memories having stored thereon a text markup language document usable by a processor, the text markup language document comprising:

a first portion identifying an input field for a form; and

a second portion identifying one or more restrictions on inputs to the input field, and further identifying validation code to be automatically added to a page to enforce the one or more restrictions on inputs to the input field.

39. (Canceled).

40. (Previously Presented) One or more computer-readable memories as recited in claim 38, wherein the first portion further identifies a type of the input field.

41. (Previously Presented) One or more computer-readable memories as recited in claim 38, wherein the second portion comprises a set of one or more attributes and, for each attribute, an associated value for the attribute.

42. (Previously Presented) One or more computer-readable memories as recited in claim 38, wherein the input field is for user-input of data.

43. (Previously Presented) One or more computer-readable memories as recited in claim 1, wherein the computer-executable instructions further direct the processor to perform acts comprising:

identifying, on the source code form definition, one or more restrictions for the custom field; and

using, in identifying the validation code, the one or more restrictions.